CONTROLLER DESIGN

ENPM667 PROJECT 2 REPORT(2 PENDULUM CART)

Akash Ravindra(117422085) and Rishabh Singh (117511208)

Maryland Robotics Center

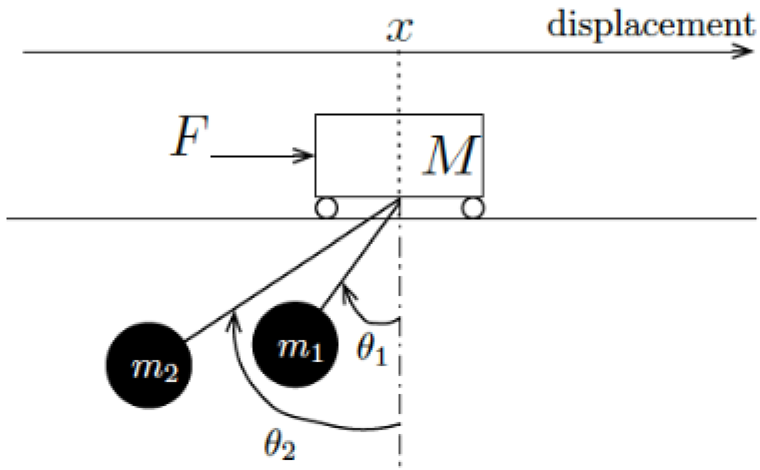
The University of Maryland, College Park

November 2021

Contents

1 INTRODUCTION

The project required us to develop the dynamics of a two-pendulum cart. We will linearize the system, implement an LQR and an LQG controller, and demonstrate its characteristics. We explore related topics such as Lyapunov stability, controllability, and observability in the process. Stated below is the system for which a controller is to be implemented.



The crane moves along an one-dimensional track. It behaves as a frictionless cart with mass M actuated. An external force F that constitutes the input of the system. There are two loads suspended from cables attached to the crane. The loads have mass m1 and m2, and the lengths of the cables are l1 and l2, respectively.

2 Equations of system

We begin by obtaining the dynamics of the system using the Lagrange mechanics. We assume the origin of the system to be at the point the masses m_1 and m_2 are suspended. The instants we analyze we assume the system has already move by a distance X along the x axis.

The position of the mass m_1 can be written as

$$\begin{aligned}
\vec{P}_{m_1,O} &= \vec{P}_{m_1,A} + \vec{P}_{A,O} \\
&= l_1 e^{-i(\frac{\pi}{2}+\theta_1)} + x \\
\text{where } e^{i\theta} &= cos\theta + i\sin\theta \\
\text{Similarly, for mass } m_2 & \\
\vec{P}_{m_2,O} &= \vec{P}_{m_2,A} + \vec{P}_{A,O} \\
&= l_2 e^{-i(\frac{\pi}{2}+\theta_2)} + x
\end{aligned} \tag{1}$$

Having found the position of the masses, we can now take its derivative with respect to time to obtain their velocities of the cart and the masses.

$$
\begin{aligned}
\vec{V}_{M,O} &= \dot{x} \\
\vec{V}_{m_1,O} &= \dot{\vec{P}}_{m_1,O} = l_1 e^{-i(\frac{\pi}{2}+\theta_1)}\dot{\theta}_1(-i) + \dot{x} \\
&= l_1\dot{\theta}_1 e^{-i\frac{\pi}{2}-i\theta_1-i\frac{\pi}{2}} + \dot{x} \\
&= l_1\dot{\theta}_1 e^{-i(\pi+\theta_1)} + \dot{x} \\
&= \dot{x} - l_1\dot{\theta}_1 e^{-i\theta_1}
\end{aligned} \tag{2}
$$

Similarly, for mass m_2

$$
\begin{aligned}
\vec{V}_{m_2,O} &= \dot{\vec{P}}_{m_2,O} = l_2 e^{-i(\frac{\pi}{2}+\theta_2)}\dot{\theta}_2(-i) + \dot{x} \\
&= \dot{x} - l_2\dot{\theta}_2 e^{-i\theta_2}
\end{aligned}
$$

Using the equations for velocity found in 2, we can obtain he kinetic energy of the whole system.

$$
\begin{aligned}
KE &= \frac{1}{2}m|\vec{V}_{MO}|^2 + \frac{1}{2}m_1|\vec{V}_{m_1,O}|^2 + \frac{1}{2}m_2|\vec{V}m_2,O|^2 \\
&= \frac{1}{2}m\dot{x}^2 + \frac{1}{2}m_1\left[(\dot{x} - l_1\dot{\theta}_1\cos\theta_1)^2 + (l_1\dot{\theta}_1\sin\theta_1)^2\right] + \frac{1}{2}m_2\left[(\dot{x} - l_2\dot{\theta}_2\cos\theta_2)^2 + (l_2\dot{\theta}_2\sin\theta_2)^2)\right] \\
&= \frac{1}{2}\dot{x}^2(m + m_1 + m_2) - \dot{x}\left[(m_1 l_1\dot{\theta}_1\cos\theta_1) + (m_2 l_2\dot{\theta}_2\cos\theta_2)\right] + \frac{1}{2}\left[m_1 l_1^2\dot{\theta}_1{}^2 + m_2 l_2^2\dot{\theta}_2{}^2\right]
\end{aligned} \tag{3}
$$

Similarly we can find the potential energy of the entire system. Note, the potential energy of the cart is zero.

$$
\begin{aligned}
PE &= PE_{m_1} + PE_{m_2} \\
&= m_1 g(-l\cos\theta_1) + m_2 g(-l_2\cos\theta_2)) \\
&= g\left[m_1 l_1\cos\theta_1 + m_2 l_2\cos\theta_2\right]
\end{aligned} \tag{4}
$$

The Lagrange is given by the difference between the kinetic energy and the potential energy

$$
\begin{aligned}
L &= KE - PE \\
&= \frac{1}{2}\dot{x}^2(m + m_1 + m_2) - \dot{x}\left[(m_1 l_1\dot{\theta}_1\cos\theta_1) + (m_2 l_2\dot{\theta}_2\cos\theta_2)\right] \\
&+ \frac{1}{2}\left[m_1 l_1^2\dot{\theta}_1{}^2 + m_2 l_2^2\dot{\theta}_2{}^2\right] \\
&- g\left[m_1 l_1\cos\theta_1 + m_2 l_2\cos\theta_2\right]
\end{aligned} \tag{5}
$$

Using the Lagrange we can now differentiate it with respect to the states.

$$
\begin{aligned}
F &= \frac{d}{dt}(\frac{\partial L}{\partial \dot{x}}) - (\frac{\partial L}{\partial x}) \\
0 &= \frac{d}{dt}(\frac{\partial L}{\partial \dot{\theta}_1}) - (\frac{\partial L}{\partial \theta_1}) \\
0 &= \frac{d}{dt}(\frac{\partial L}{\partial \dot{\theta}_2}) - (\frac{\partial L}{\partial \theta_2})
\end{aligned} \tag{6}
$$

Simplifying the above equation we get the following.

$$
F = [M + m_1 + m_2]\ddot{x} - m_1 l_1\ddot{\theta}_1\cos(\theta_1) + m_1 l_1\dot{\theta}_1{}^2\sin(\theta_1) - m_2 l_2\ddot{\theta}_2\cos(\theta_2) + m_2 l_2\dot{\theta}_2{}^2\sin(\theta_2) \tag{7}
$$

$$
0 = m_1 l_1^2\ddot{\theta}_1 - m_1\ddot{x}l_1\cos(\theta_1) + m_1 l_1 g\sin(\theta_1) \tag{8}
$$

$$
0 = m_2 l_2^2\ddot{\theta}_2 - m_2\ddot{x}l_2\cos(\theta_2) + m_2 l_2 g\sin(\theta_2) \tag{9}
$$

3 Linearization of system

The equations 7,8 and 9 that were found in the previous section are all Non linear. So as expected we linearize the system of equation about the given point. $x = \theta_1 = \theta_2 = 0$.

In order to construct the state space representation for the set of equations, we can choose the follow variables to be our state variables (in the same order)

$$[x; \dot{x}; \theta_1; \dot{\theta_1}; \theta_2; \dot{\theta_2}] \tag{10}$$

Given the standard form of the state space equation as,

$$X = A_F X + B_F U \tag{11}$$

We can find A_F around the equilibrium point using the Jacobian matrix

$$A_F = \begin{bmatrix} \frac{\partial F_1}{\partial x} & \frac{\partial F_1}{\partial \dot{x}} & \frac{\partial F_1}{\partial \theta_1} & \frac{\partial F_1}{\partial \dot{\theta_1}} & \frac{\partial F_1}{\partial \theta_2} & \frac{\partial F_1}{\partial \theta_2} \\ \frac{\partial F_2}{\partial x} & \frac{\partial F_2}{\partial \dot{x}} & \frac{\partial F_2}{\partial \theta_1} & \frac{\partial F_2}{\partial \dot{\theta_1}} & \frac{\partial F_2}{\partial \theta_2} & \frac{\partial F_2}{\partial \theta_2} \\ \frac{\partial F_3}{\partial x} & \frac{\partial F_3}{\partial \dot{x}} & \frac{\partial F_3}{\partial \theta_1} & \frac{\partial F_3}{\partial \dot{\theta_1}} & \frac{\partial F_3}{\partial \theta_2} & \frac{\partial F_3}{\partial \theta_2} \\ \frac{\partial F_4}{\partial x} & \frac{\partial F_4}{\partial \dot{x}} & \frac{\partial F_4}{\partial \theta_1} & \frac{\partial F_4}{\partial \dot{\theta_1}} & \frac{\partial F_4}{\partial \theta_2} & \frac{\partial F_4}{\partial \theta_2} \\ \frac{\partial F_5}{\partial x} & \frac{\partial F_5}{\partial \dot{x}} & \frac{\partial F_5}{\partial \theta_1} & \frac{\partial F_5}{\partial \dot{\theta_1}} & \frac{\partial F_5}{\partial \theta_2} & \frac{\partial F_5}{\partial \theta_2} \\ \frac{\partial F_6}{\partial x} & \frac{\partial F_6}{\partial \dot{x}} & \frac{\partial F_6}{\partial \theta_1} & \frac{\partial F_6}{\partial \dot{\theta_1}} & \frac{\partial F_6}{\partial \theta_2} & \frac{\partial F_6}{\partial \theta_2} \end{bmatrix} \tag{12}$$

Where F_1 = $\dot{x}$,F_2 = Eq 7, F_3 = $\dot{\theta_1}$, F_4 = Eq 8, F_5 = $\dot{\theta_2}$, F_6 = Eq 9

Simplifying the we get A_F as

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{-m_1 g}{M} & 0 & \frac{-m_2 g}{M} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{(M+m_1)g}{Ml_1} & 0 & \frac{-m_2 g}{Ml_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{-m_1 g}{Ml_2} & 0 & \frac{(M+m_2)g}{Ml_2} & 0 \end{bmatrix} \tag{13}$$

Similarly B_F can be found using the Jacobian matrix

$$B_F = \begin{bmatrix} \frac{\partial U_1}{\partial x} \\ \frac{\partial F_2}{\partial \dot{x}} \\ \frac{\partial F_3}{\partial \theta_1} \\ \frac{\partial F_4}{\partial \dot{\theta_1}} \\ \frac{\partial F_5}{\partial \theta_2} \\ \frac{\partial F_6}{\partial \dot{\theta_2}} \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{M} \\ 0 \\ \frac{1}{Ml_1} \\ 0 \\ \frac{1}{Ml_2} \end{bmatrix} \tag{14}$$

4 Controllability check

C. Controllability of state space model

```
clc
clear all
close all
syms M m1 m2 l1 l2 g;
```

A and B matrices of the linearized state space model

```
A = [0 1 0 0 0 0;
    0 0 -(m1*g)/M 0 -(m2*g)/M 0;
    0 0 0 1 0 0;
    0 0 -((M+m1)*g)/(M*l1) 0 -(m2*g)/(M*l1) 0;
    0 0 0 0 0 1;
    0 0 -(m1*g)/(M*l2) 0 -(g*(M+m2))/(M*l2) 0];

B = [0; 1/M; 0; 1/(M*l1); 0; 1/(M*l2)];
```

Checking for controllability

Controllability matrix

```
c_mat = [B A*B (A^2)*B (A^3)*B (A^4)*B (A^5)*B];
num_uncon_states = length(A) - rank(c_mat);

% Right now the system is controllable with zero uncontrollable states

%% Checking for conditions in which the system isn't controllable

det_c_mat = simplify(det(c_mat));
```

The determinant comes out to be -(g^6*(l1 - l2)^2)/(M^6*l1^6*l2^6).The determinant would result in zero for l1 = l2, or would tend towardszero for very high value of M, l1 and l2 which is pratically not feasible, hence the only possible way for the system to become uncontrollable is l1=l2

```
%% Checking for the number of uncontrollable states when l1=l2
% let us substitute l1=l2

c_mat = subs(c_mat,l1,l2);
det_c_mat = simplify(det(c_mat))
```

det_c_mat = 0

```
num_uncon_states_new = length(A) - rank(c_mat)
```

num_uncon_states_new = 2

The determinant becomes zero when $l_1 = l_2$

Number of uncontrollable states = 2

5 LQR

D. Design of a LQR controller

```
clc
clear all
close all
```

Parameters and their values

```
M=1000;     % Cart
m1=100;     % Bob 1
m2=100;     % Bob 2
l1=20;      % Bob 1 string length
l2=10;      % Bob 2 string length
g=9.81;     % Acceleration due to gravity
```

A and B matrices for the state space model

```
A = [0 1 0 0 0 0;
    0 0 -(m1*g)/M 0 -(m2*g)/M 0;
    0 0 0 1 0 0;
    0 0 -((M+m1)*g)/(M*l1) 0 -(m2*g)/(M*l1) 0;
    0 0 0 0 0 1;
    0 0 -(m1*g)/(M*l2) 0 -(g*(M+m2))/(M*l2) 0];

B = [0; 1/M; 0; 1/(M*l1); 0; 1/(M*l2)];
```

Assuming the outputs the same as the states, the C and D matrix of the state space model turn out to be :

```
C = eye(size(A,1));
D = zeros(size(A,1),1);

% Defining state space model for given A,B,C,D
sys = ss(A,B,C,D);

%% Controllability of the system

num_uncon_states = length(A) - rank(ctrb(A,B))

%% LQR controller design

% states : Y = [X, X_dot, theta1, theta1_dot, theta2, theta2_dot

% Initial values for the states
Y_initial = [5;0;20;0;40;0];
```

Case 1 : weights = 1

```
% Initializing the initial weights in Q and R as 1
Q=1*eye(size(A,1));
R=1;
```

```
figure
initial(sys,Y_initial)
grid on   %grid lines visible

[K_lqr, P, poles] = lqr(A,B,Q,R);

sys = ss(A-(B*K_lqr),B,C,D);
figure
initial(sys,Y_initial)
grid on
```

Case 2 : Q_weights = 1000, R = 1

```
% Initializing the initial weights in Q and R as 1
Q=1000*eye(size(A,1));
R=1;

figure
initial(sys,Y_initial)
grid on

[K_lqr, P, poles] = lqr(A,B,Q,R);

sys = ss(A-(B*K_lqr),B,C,D);
figure
initial(sys,Y_initial)
grid on
```

Case 3 : Q_weights = 1000, R = 0.1

```
% Initializing the initial weights in Q and R as 1
Q=1000*eye(size(A,1));
R=0.1;

figure
initial(sys,Y_initial)
grid on

[K_lqr, P, poles] = lqr(A,B,Q,R);

sys = ss(A-(B*K_lqr),B,C,D);
figure
initial(sys,Y_initial)
grid on
```

Case 4 : R = 0.1

```
% Initializing the initial weights in Q and R as 1
Q=100*eye(size(A,1));
R=0.01;
```

Penalizing theta1, theta2 and its derivatives

```
Q(4,4) = 100000;
Q(6,6) = 100000;

figure
initial(sys,Y_initial)
grid on

[K_lqr, P, poles] = lqr(A,B,Q,R);

sys = ss(A-(B*K_lqr),B,C,D);
figure
initial(sys,Y_initial)
grid on


% Increasing the value of Q weights decreases the convergence time for the
% states while increasing the R weights decreases the frequency of
% oscillations or perturbation in the system. Hence, we choose
```

Part D(b) - Non-Linear Calculations

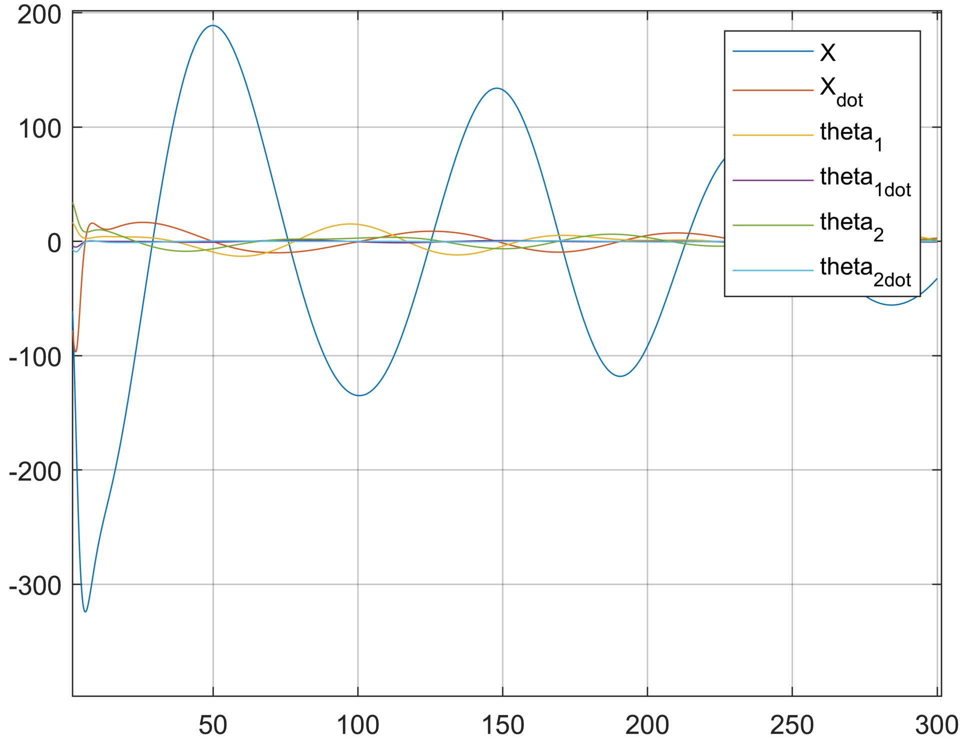
```
clc
clear all
close all
```

Using optimal gain K we got by solving for lqr controller for thelinearized state space model

```
K_lqr = [99.9999999999980,545.131345054836,
    553.135499158470,2680.31266537002,
    1760.46389471323,1824.83063433848];


y_init = [5; 0; 20; 0; 40; 0]
```

```
y_init = 6×1
     5
     0
    20
     0
    40
     0
```

```
tspan = 0:0.1:300;
[t,y] = ode45(@pendulum_cart,tspan,y_init);
plot(t,y)
grid on
legend('X','X_{dot}','theta_1','theta_1_{dot}','theta_2','theta_2_{dot}');
```




```
function dydt = pendulum_cart(t,y)

% Parameters and their values

M=1000;     % Cart
m1=100;     % Bob 1
m2=100;     % Bob 2
l1=20;      % Bob 1 string length
l2=10;      % Bob 2 string length
g=9.81;     % Acceleration due to gravity

K_lqr = [99.9999999999980,545.131345054836,
    553.135499158470,2680.31266537002,
    1760.46389471323,1824.83063433848];
F= -K_lqr*y;
dydt=zeros(6,1);

dydt(1) = y(2);
dydt(2)=(F-(g/2)*(m1*sind(2*y(3))+m2*sind(2*y(5)))- ...
    (m1*l1*(y(4)^2)*sind(y(3)))- ...
    (m2*l2*(y(6)^2)*sind(y(5))))/(M+m1*((sind(y(3)))^2)+m2*((sind(y(5)))^2));
dydt(3)= y(4);
dydt(4)= (dydt(2)*cosd(y(3))-g*(sind(y(3))))/l1';
dydt(5)= y(6);
dydt(6)= (dydt(2)*cosd(y(5))-g*(sind(y(5))))/l2;
```

```
end
```

6 Observer

E. Observabilty of the linearized state

```
clc
clear all
close all
```

Parameters and their values

```
M=1000;     % Cart
m1=100;     % Bob 1
m2=100;     % Bob 2
l1=20;      % Bob 1 string length
l2=10;      % Bob 2 string length
g=9.81;     % Acceleration due to gravity
```

A and B matrices for the state space model

```
A = [0 1 0 0 0 0;
    0 0 -(m1*g)/M 0 -(m2*g)/M 0;
    0 0 0 1 0 0;
    0 0 -((M+m1)*g)/(M*l1) 0 -(m2*g)/(M*l1) 0;
    0 0 0 0 0 1;
    0 0 -(m1*g)/(M*l2) 0 -(g*(M+m2))/(M*l2) 0];

%B = [0; 1/M; 0; 1/(M*l1); 0; 1/(M*l2)];

% Assuming the outputs the same as the states, the C and D matrix of the
% state space model turn out to be :

% Defining state space model for given A,B,C,D
%sys = ss(A,B,C,D);
```

Observability of the system

```
% Only x(t)
C1 = [1 0 0 0 0 0];

disp('Number of unobservable states when only x(t) is observed: ')
```

```
Number of unobservable states when only x(t) is observed:
```

```
num_unobs_states_cond1 = length(A) - rank(obsv(A,C1))
```

```
num_unobs_states_cond1 = 0
```

```
% Only theta1(t) and theta2(t)
C2 = [0 0 1 0 0 0; 0 0 0 1 0 0];

disp('Number of unobservable states when only theta1(t) and theta2(t) are observed: ')
```

```
Number of unobservable states when only theta1(t) and theta2(t) are observed:
```

```
num_unobs_states_cond2 = length(A) - rank(obsv(A,C2))
```

```
num_unobs_states_cond2 = 2
```

```
% Only x(t) and theta2(t)
C3 = [0 1 0 0 0 0; 0 0 0 1 0 0];

disp('Number of unobservable states when only x(t) and theta2(t) are observed: ')
```

```
Number of unobservable states when only x(t) and theta2(t) are observed:
```

```
num_unobs_states_cond3 = length(A) - rank(obsv(A,C3))
```

```
num_unobs_states_cond3 = 1
```

```
% Only x(t), theta1(t) and theta2(t)
C4 = [1 0 0 0 0 0; 0 0 1 0 0 0;0 0 0 0 1 0];

disp('Number of unobservable states when only x(t), theta1(t) and theta2(t) are observed: ')
```

```
Number of unobservable states when only x(t), theta1(t) and theta2(t) are observed:
```

```
num_unobs_states_cond4 = length(A) - rank(obsv(A,C4))
```

```
num_unobs_states_cond4 = 0
```

```
C5 = eye(size(A,1));

disp('Number of unobservable states when all states are observed: ')
```

```
Number of unobservable states when all states are observed:
```

```
num_unobs_states_cond5 = length(A) - rank(obsv(A,C5))
```

```
num_unobs_states_cond5 = 0
```

F. Luenberger Observer for linearized state space

```
clc
clear all
close all

% Given Parameters and their values

M=1000;     % Cart
m1=100;     % Bob 1
m2=100;     % Bob 2
l1=20;      % Bob 1 string length
l2=10;      % Bob 2 string length
g=9.81;     % Acceleration due to gravity
```

Linearized matrix is given as follows

```
A=[0 1 0 0 0 0; 0 0 -(m1*g)/M 0 -(m2*g)/M 0; 0 0 0 1 0 0;
    0 0 -((M+m1)*g)/(M*l1) 0 -(m2*g)/(M*l1) 0; 0 0 0 0 0 1;
    0 0 -(m1*g)/(M*l2) 0 -(g*(M+m2))/(M*l2) 0];
```

Finding B and C matrices according to observability found in prob e

```
B=[0; 1/M; 0; 1/(M*l1); 0; 1/(M*l2)];

C1 = [1 0 0 0 0 0];  % X1
C3 = [1 0 0 0 0 0; 0 0 0 0 1 0]; %X1 and theta2
C4 = [1 0 0 0 0 0;0 0 1 0 0 0; 0 0 0 0 1 0]; %X1, theta1, theta2
```

D is assumed to be zero and Q is retuned according to the response

```
D = 0;

% Q and R matrices
```

```
Q=[100 0 0 0 0 0;
    0 100 0 0 0 0;
    0 0 100000 0 0 0;
    0 0 0 100000 0 0;
    0 0 0 0 100000 0;
    0 0 0 0 0 100000];


% Luenberger Observer with initial condition for state variables

Y_init = [0,0,20,0,40,0,0,0,0,0,0,0];
```

Forced Pole placement for stability of A-LC

```
% Eigen values for pole placement ; -ve real part is considered
po = [-1;-2.5;-3.2;-4.6;-5;-8];
% Finding L gain using pole placement
L1 = place(A',C1',po)' ;
L3 = place(A',C3',po)' ;
L4 = place(A',C4',po)' ;
```

Finding controller matrix

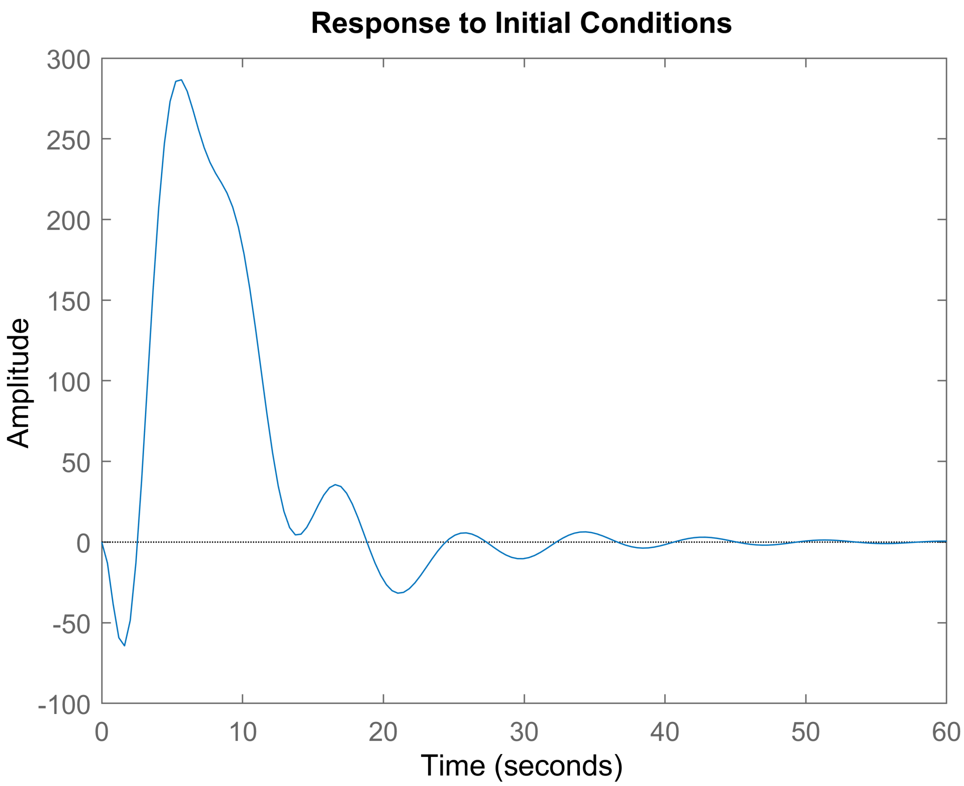
```
R=0.01;
K=lqr(A,B,Q,R);
```

Defining the Augmented matrix given by seperation principle

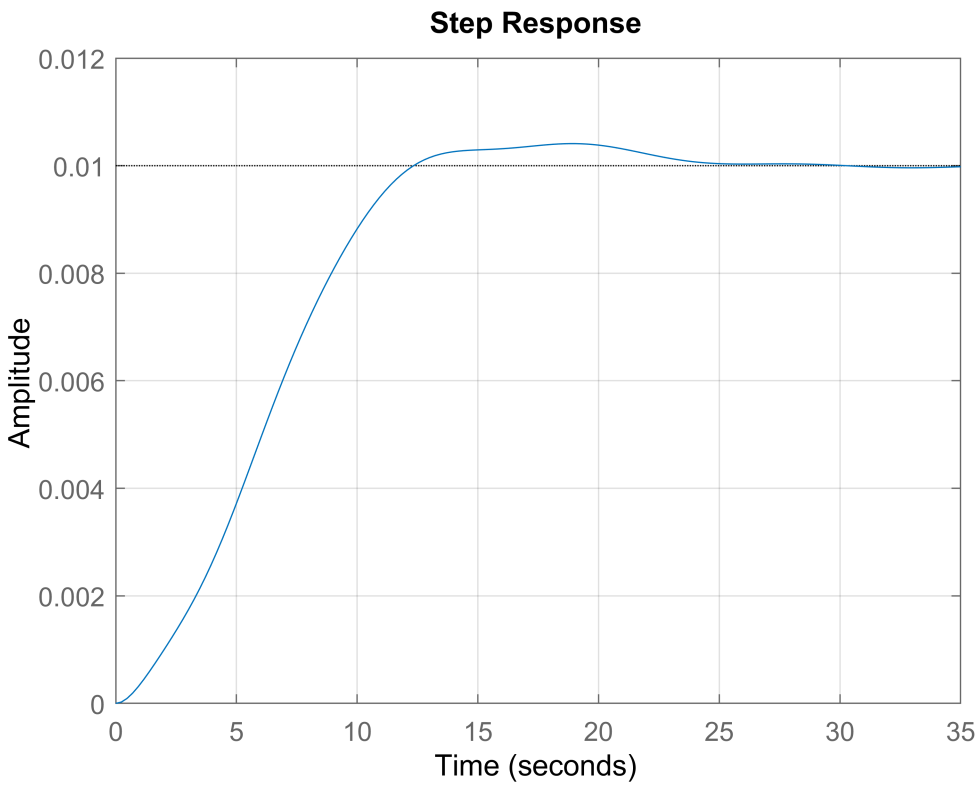
```
ABLC_1 = [(A-B*K) B*K; zeros(size(A)) (A-L1*C1)];
Bc = [B;zeros(size(B))];
Cc1 = [C1 zeros(size(C1))];

% for C3, Augmented matrix
ABLC_3 = [(A-B*K) B*K;
    zeros(size(A)) (A-L3*C3)];
Cc3 = [C3 zeros(size(C3))];% Luenberger C matrix

% for C4, Augmented matrix
ABLC_4 = [(A-B*K) B*K;% Luenberger A matrix
    zeros(size(A)) (A-L4*C4)];
Cc4 = [C4 zeros(size(C4))];% Luenberger C matrix
```

Step and Initial response

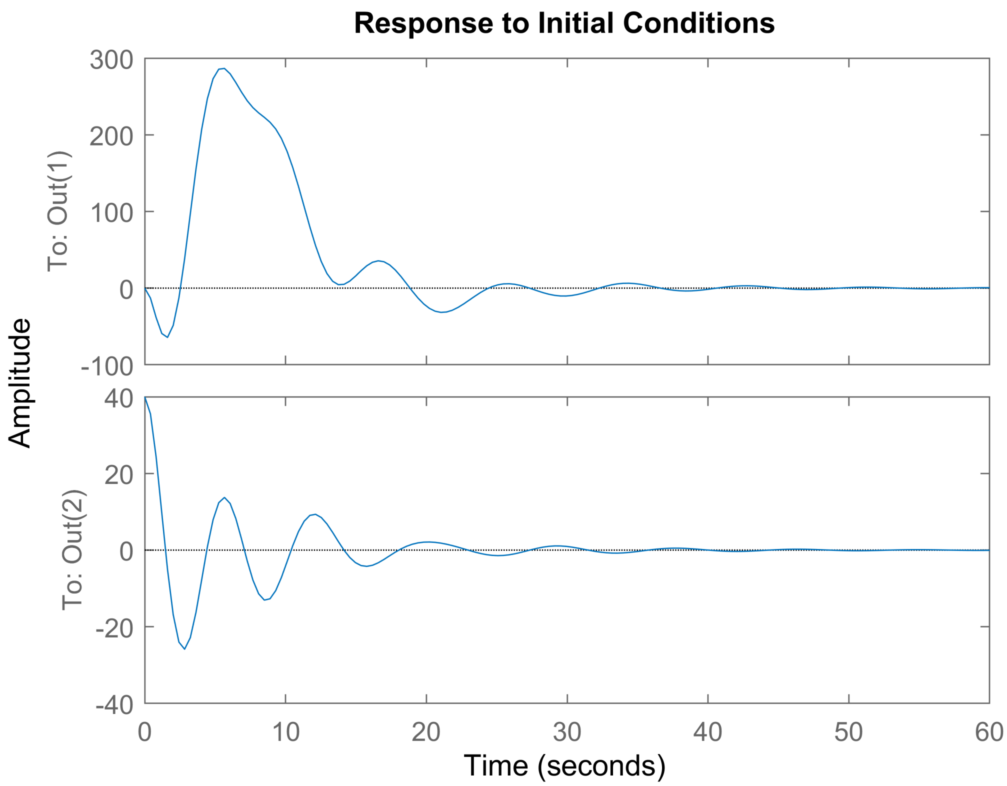
```
sys_1 = ss(ABLC_1, Bc, Cc1,D);
figure
initial(sys_1,Y_init)
```



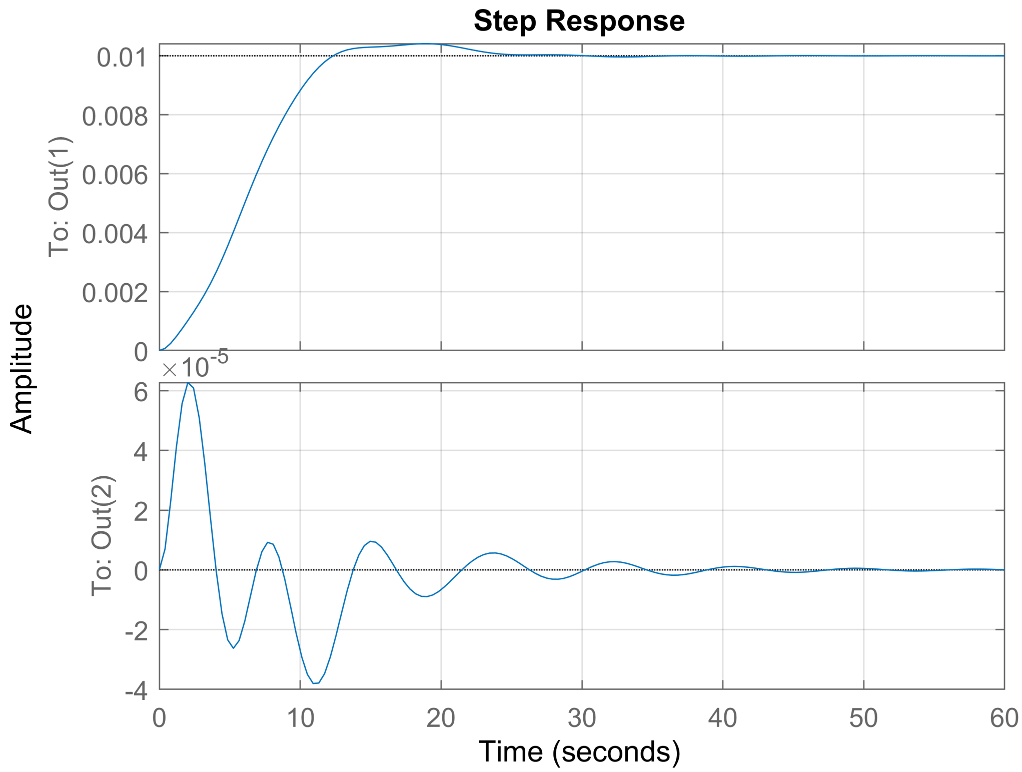

```
figure
step(sys_1)
grid on
```



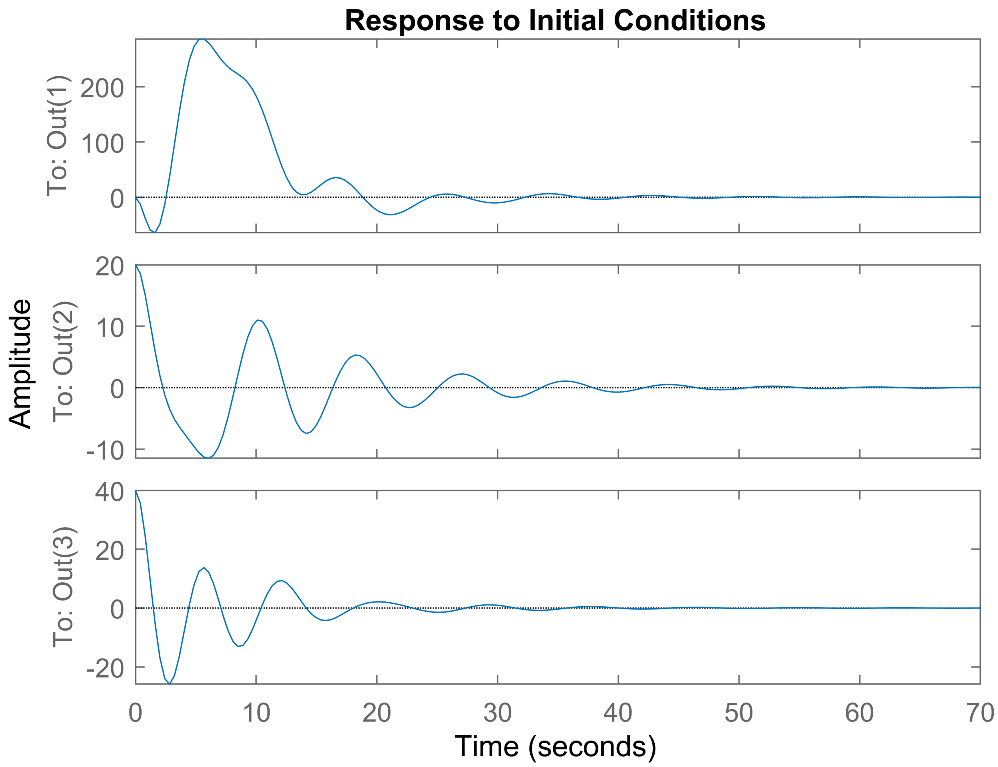

```
sys_2 = ss(ABLC_3, Bc, Cc3,D);
figure
initial(sys_2,Y_init)
```



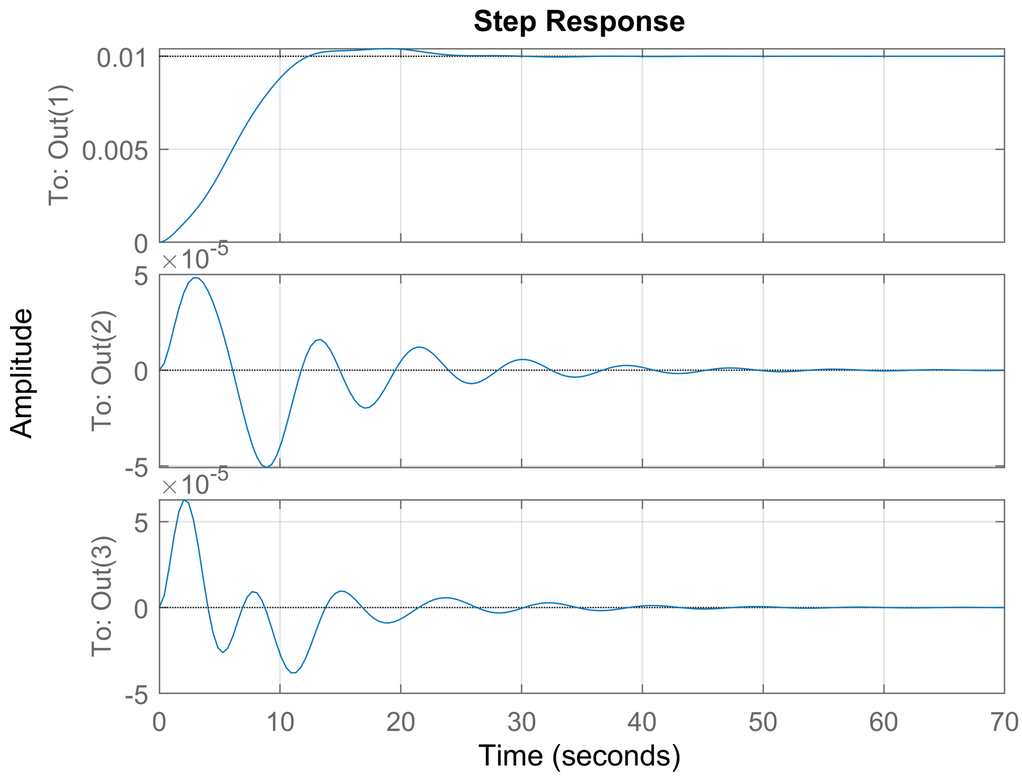

```
figure
step(sys_2)
grid on
```




```
sys_3 = ss(ABLC_4, Bc, Cc4,D);
figure
initial(sys_3,Y_init)
```




```
figure
step(sys_3)
grid on
```


Step Response
Amplitude
To: Out(1)
0.01
0.005
0
To: Out(2)
×10^-5
5
0
-5
To: Out(3)
×10^-5
5
0
-5
0
10
20
30
40
50
60
70
Time (seconds)

7 LQG Controller

G. LQG - Linear Part

```
clc
clear all
close all
```

Parameters and values

```
M = 1000;
m1 = 100;
m2 = 100;
l1 = 20;
l2 = 10;
g = 9.81;
```

A and B matrices

```
A=[0 1 0 0 0 0; 0 0 -(m1*g)/M 0 -(m2*g)/M 0;
    0 0 0 1 0 0; 0 0 -((M+m1)*g)/(M*l1) 0 -(m2*g)/(M*l1) 0;
    0 0 0 0 0 1; 0 0 -(m1*g)/(M*l2) 0 -(g*(M+m2))/(M*l2) 0];

B=[0; 1/M; 0; 1/(M*l1); 0; 1/(M*l2)];
```

Q and R matrices

```
Q = [100 0 0 0 0 0;
   0 100 0 0 0 0;
   0 0 100 0 0 0;
   0 0 0 100000 0 0;
   0 0 0 0 100 0;
   0 0 0 0 0 100000];

R = 0.01;
```

Considering C1, C2 and C3 as the states were observable

For X

```
C1 = [1 0 0 0 0 0];
```

For X and theta2

```
C3 = [1 0 0 0 0 0; 0 0 0 0 1 0];
```

For X, theta1, theta2

```
C4 = [1 0 0 0 0 0; 0 0 1 0 0 0; 0 0 0 0 1 0];

D = 0;
```

Initial condition for state variables

```
x0 = [5;0;20;0;40;0;0;0;0;0;0;0];
```

Calculating K matrix

```
K =lqr(A,B,Q,R);
```

Process noise

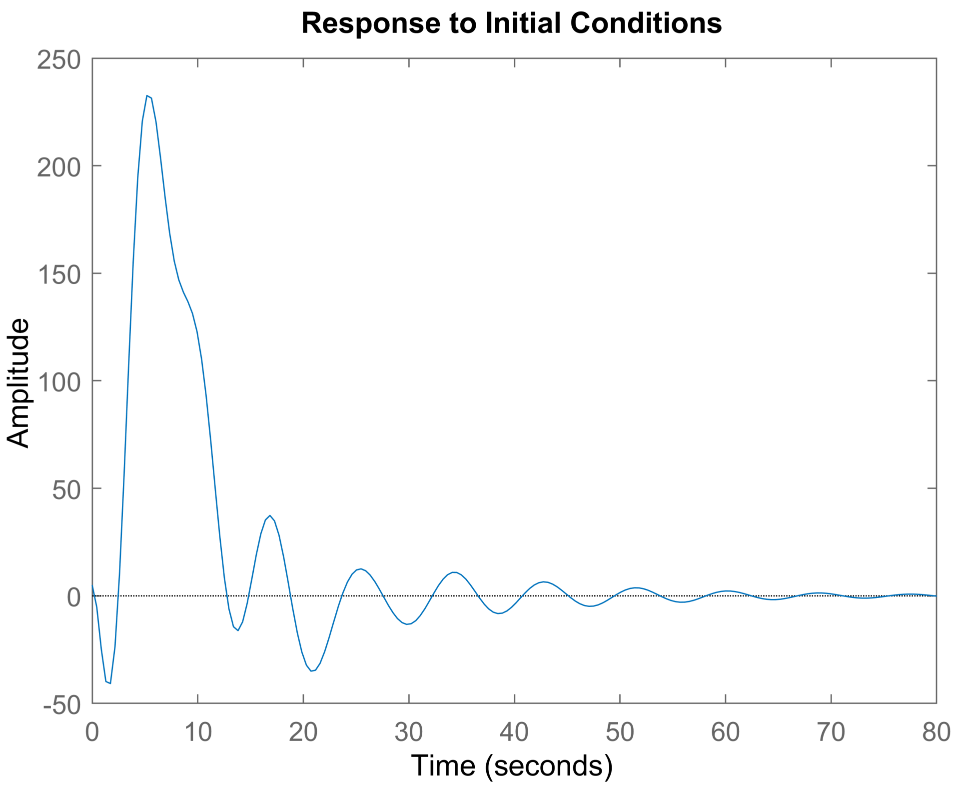
```
v_p=0.5*eye(6);
```

Measurement noise for estimation/error matrix and gain

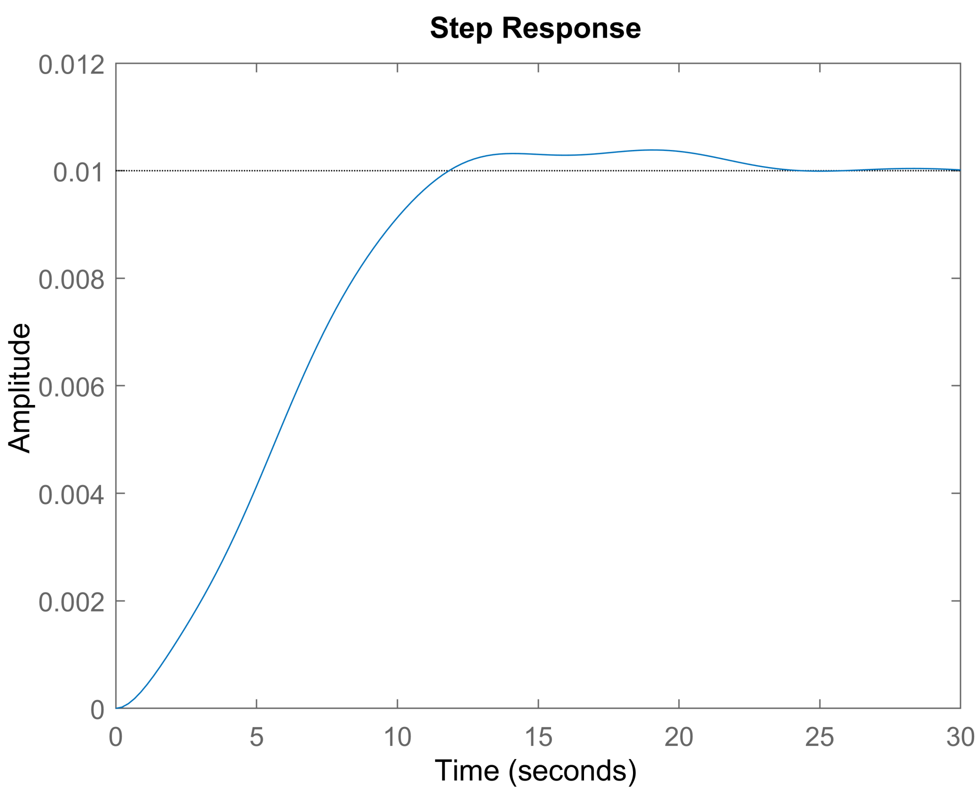
```
v_m=1;

K_kal1=lqr(A',C1',v_p,v_m)';
K_kal3=lqr(A',C3',v_p,v_m)';
K_kal4=lqr(A',C4',v_p,v_m)';
```

Observing state space considering C1

```
sys1 = ss([(A-B*K) B*K; zeros(size(A)) (A-K_kal1*C1)], [B;zeros(size(B))],[C1 zeros(size(C1))]
figure
initial(sys1,x0)
```

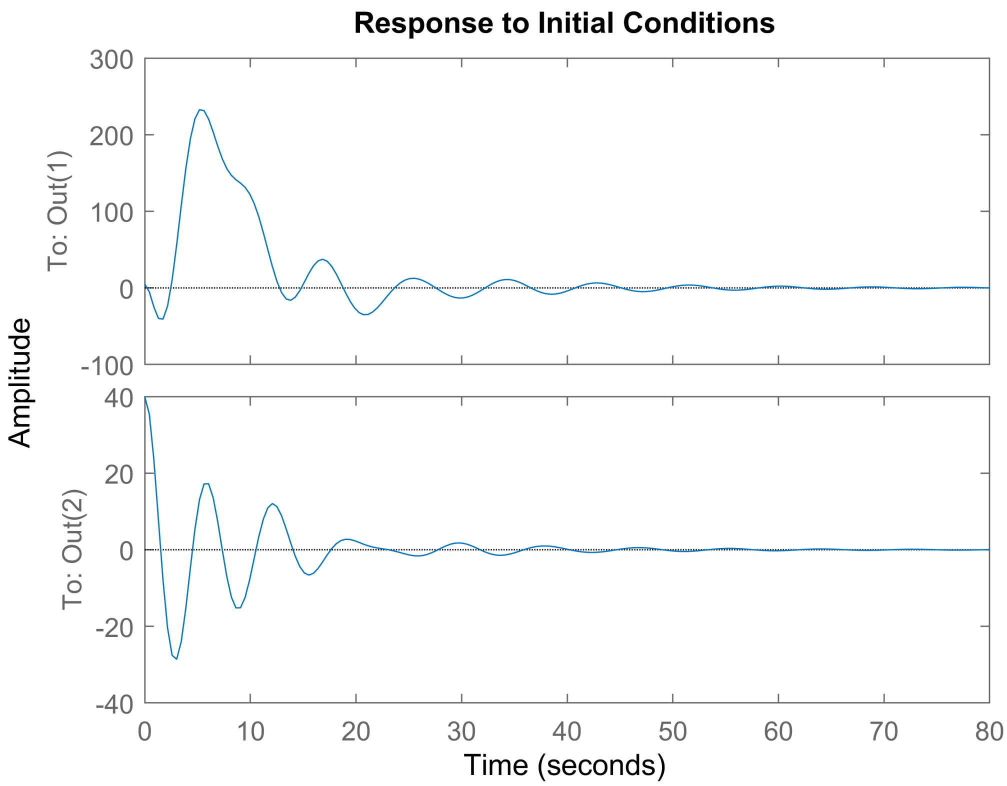



```
figure
step(sys1)
```

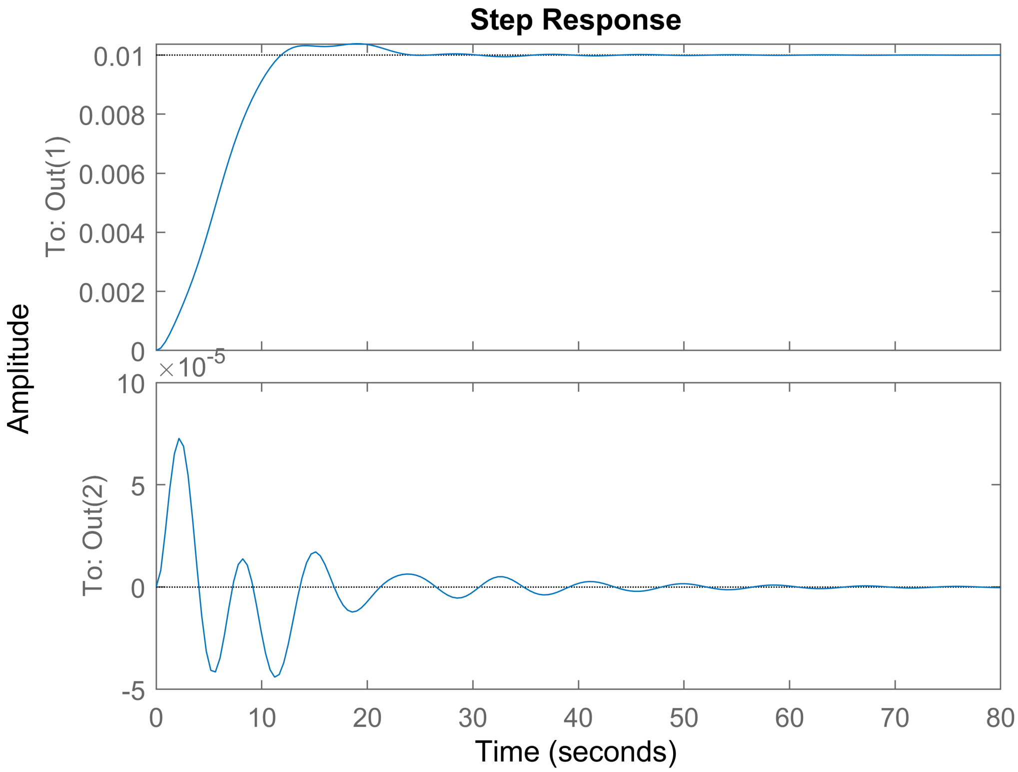


Observing using C3

```
sys3 = ss([(A-B*K) B*K; zeros(size(A)) (A-K_kal3*C3)], [B;zeros(size(B))],[C3 zeros(size(C3))],
figure
initial(sys3,x0)
```

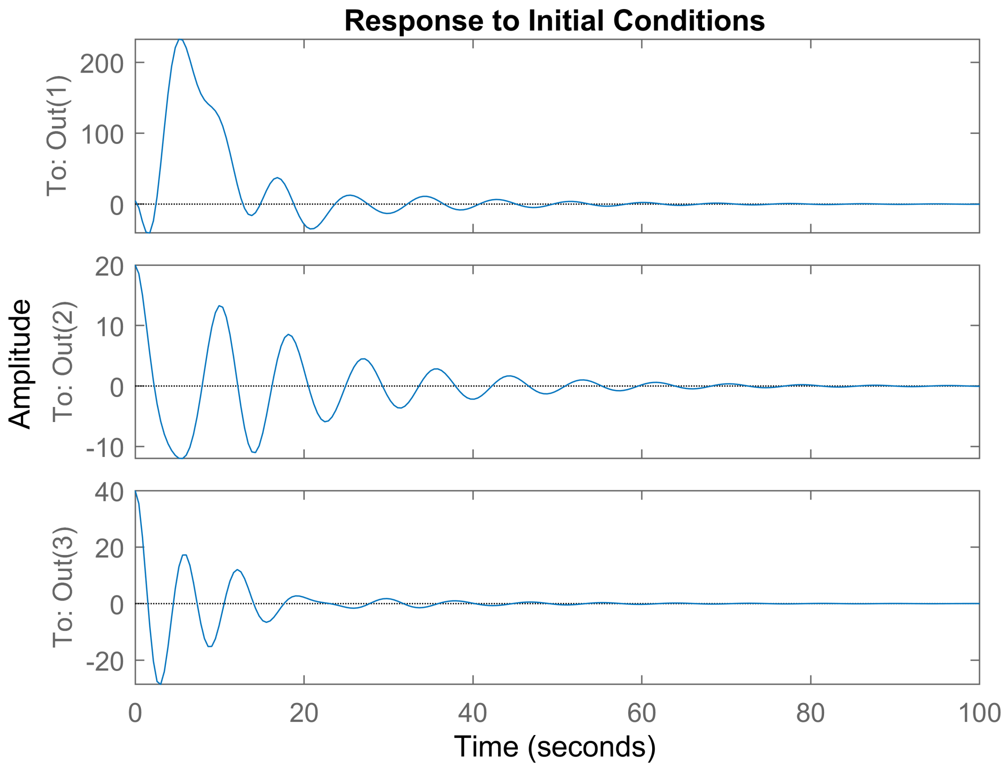



```
figure
step(sys3)
```

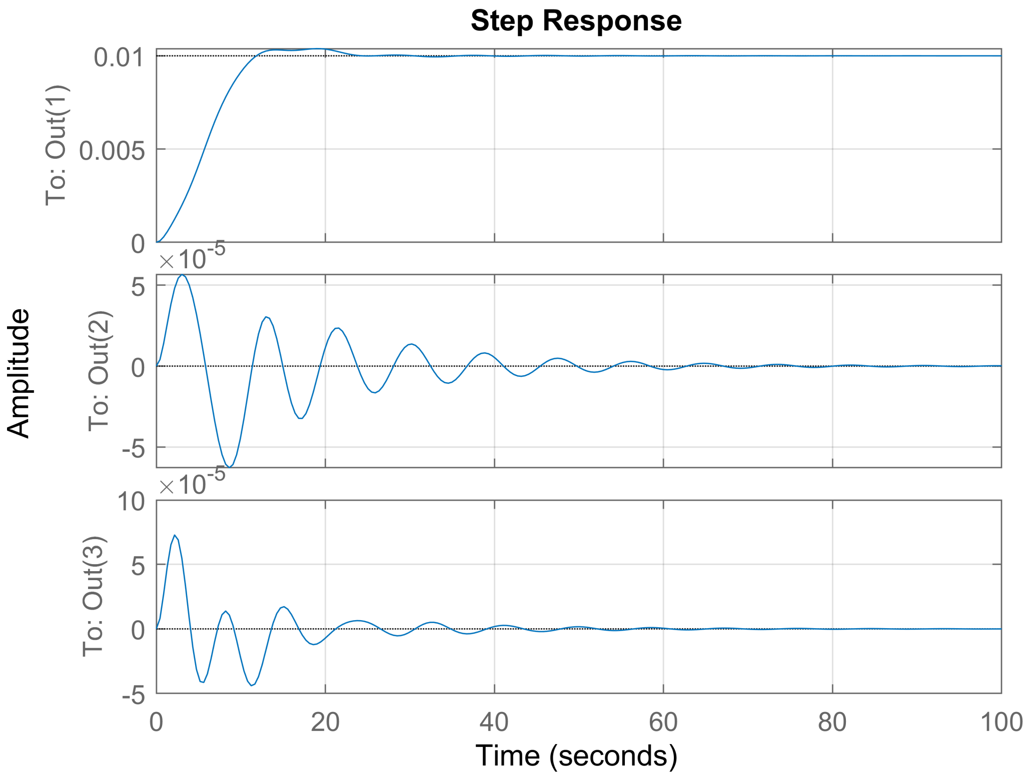


Observing using C4

```
sys4 = ss([(A-B*K) B*K; zeros(size(A)) (A-K_kal4*C4)], [B;zeros(size(B))],[C4 zeros(size(C4))]
figure
initial(sys4,x0)
```




```
figure
step(sys4)

grid on
```



Step Response
Amplitude
To: Out(1)
0.01
0.005
0
To: Out(2)
×10^-5
5
0
-5
To: Out(3)
×10^-5
10
5
0
-5
0
20
40
60
80
100
Time (seconds)

8 Conclusion

We developed the dynamics of the cart with two pendulums, following which we obtained a linearized about the given equilibrium points. We went forward in calculating the jacobians; After defining the A_F and B_F matrix, we figured the controllability of the linear system. We designed an LQR controller, Q and R values that provide good responses and demonstrate the system's response for different values of Q and R. Accordingly; we penalized our state variables

We check if the given set of variables is observable for the output feedback. We obtained three different C matrices, and then we went on to calculate the augmented AC matrix using state feedback and L. We tried to tune our variable such that the system was stable. Without a benchmark to gauge our response, we could not verify if this was an ideal response of the system.